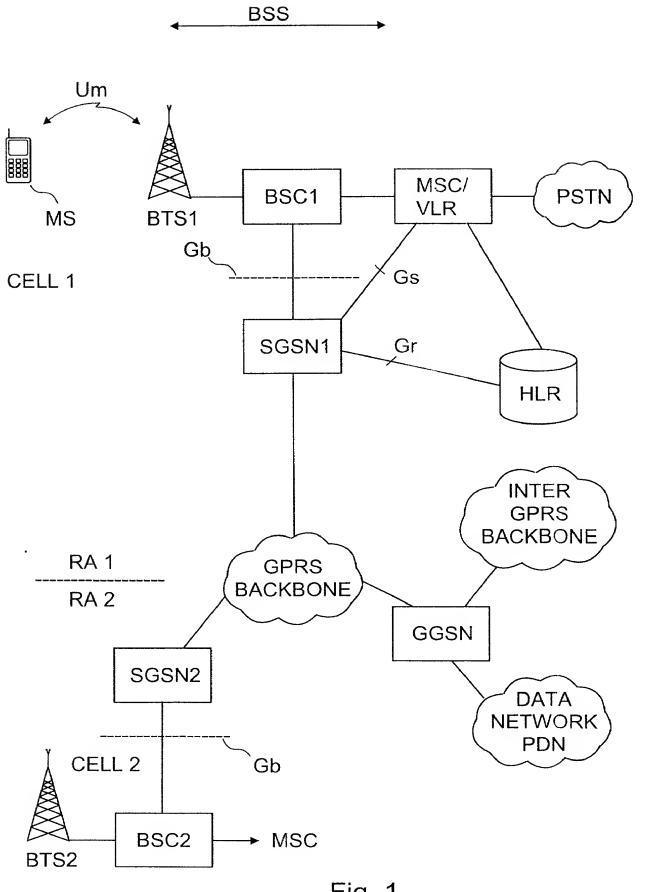
BSS
Um
MS
BTS1
BSC1
MSC/
VLR
PSTN
Gb
Gs
CELL 1
SGSN1
Gr
HLR
INTER
GPRS
BACKBONE
RA 1
RA 2
GPRS
BACKBONE
GGSN
SGSN2
DATA
NETWORK
PDN
CELL 2
Gb
BSC2
MSC
BTS2

Fig. 1

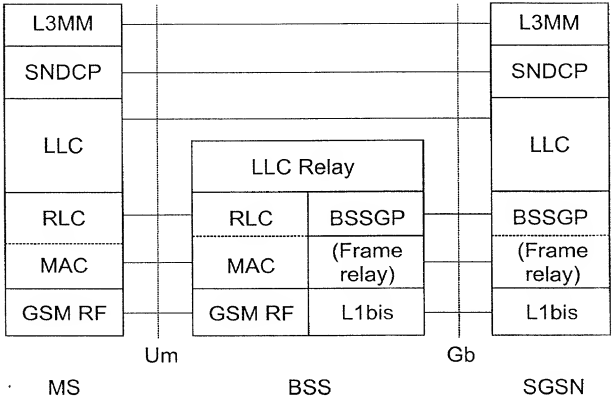
L3MM
SNDCP
LLC
RLC
MAC
GSM RF
LLC Relay
RLC
BSSGP
MAC
(Frame relay)
GSM RF
L1bis
L3MM
SNDCP
LLC
BSSGP
(Frame relay)
L1bis
Um
Gb
MS
BSS
SGSN

Fig. 2

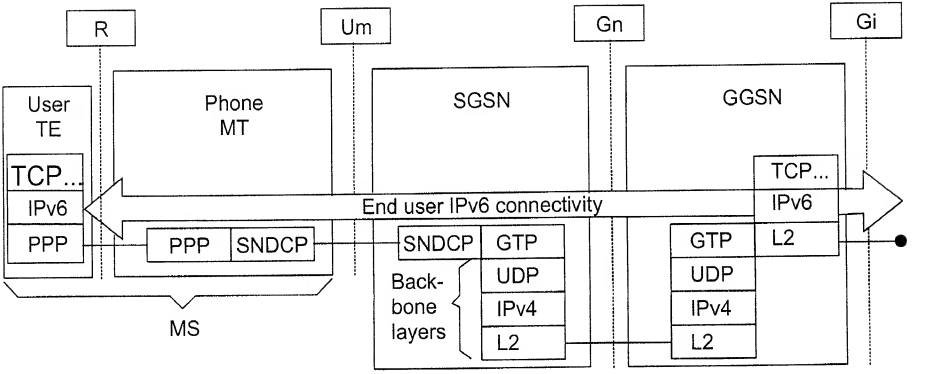
R
Um
Gn
Gi
User TE
Phone MT
SGSN
GGSN
TCP...
IPv6
PPP
End user IPv6 connectivity
PPP
SNDCP
SNDCP
GTP
UDP
IPv4
L2
Back-bone layers
GTP
UDP
IPv4
L2
TCP...
IPv6
L2
MS

Fig. 3

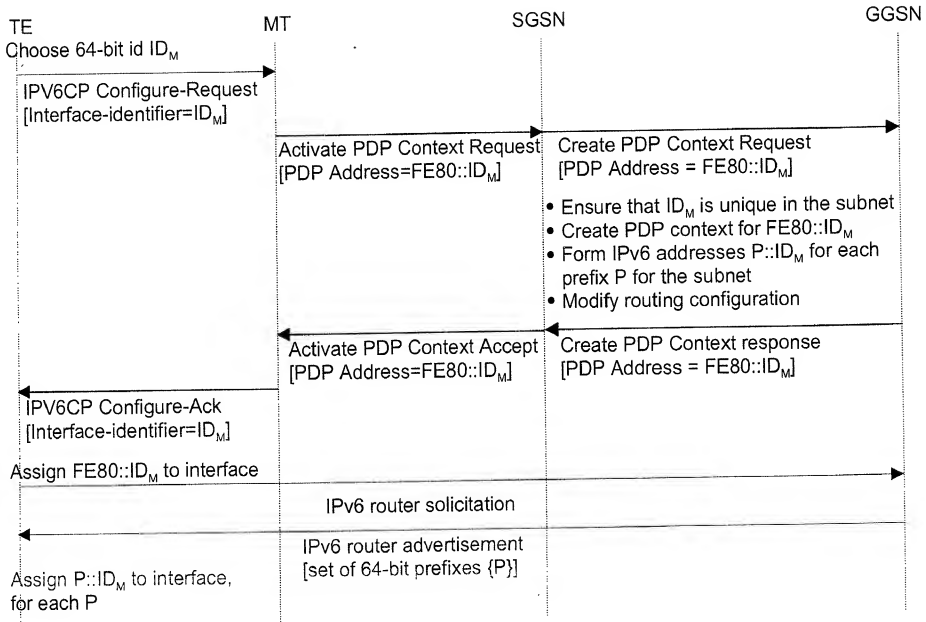
TE
MT
SGSN
GGSN
Choose 64-bit id ID_M
IPV6CP Configure-Request
[Interface-identifier=ID_M]
Activate PDP Context Request
[PDP Address=FE80::ID_M]
Create PDP Context Request
[PDP Address = FE80::ID_M]
• Ensure that ID_M is unique in the subnet
• Create PDP context for FE80::ID_M
• Form IPv6 addresses P::ID_M for each prefix P for the subnet
• Modify routing configuration
Activate PDP Context Accept
[PDP Address=FE80::ID_M]
Create PDP Context response
[PDP Address = FE80::ID_M]
IPV6CP Configure-Ack
[Interface-identifier=ID_M]
Assign FE80::ID_M to interface
IPv6 router solicitation
IPv6 router advertisement
[set of 64-bit prefixes {P}]
Assign P::ID_M to interface,
for each P

Fig. 4

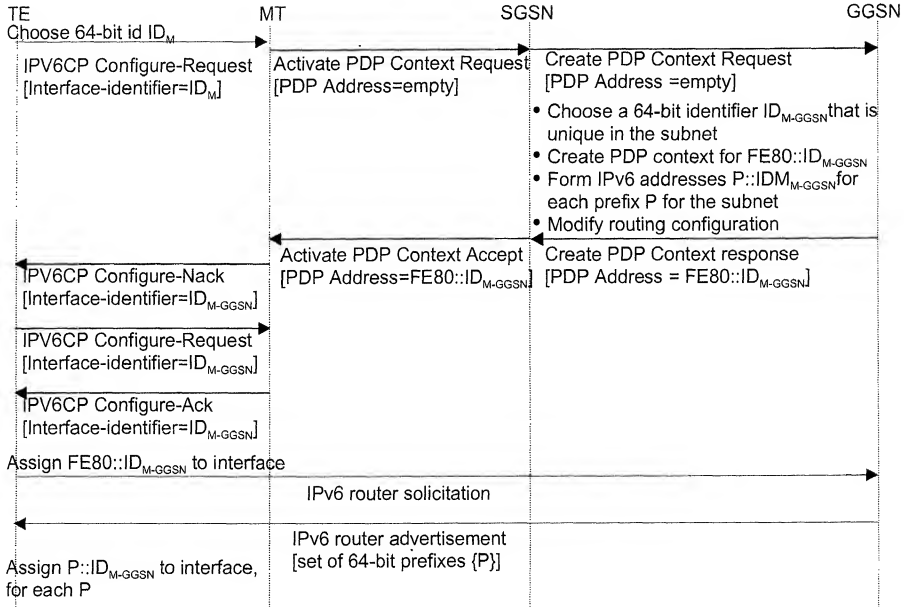
TE
MT
SGSN
GGSN
Choose 64-bit id ID_M
IPV6CP Configure-Request
[Interface-identifier=ID_M]
Activate PDP Context Request
[PDP Address=empty]
Create PDP Context Request
[PDP Address =empty]
• Choose a 64-bit identifier ID_{M-GGSN} that is unique in the subnet
• Create PDP context for FE80::ID_{M-GGSN}
• Form IPv6 addresses P::IDM$_{M-GGSN}$ for each prefix P for the subnet
• Modify routing configuration
Activate PDP Context Accept
[PDP Address=FE80::ID_{M-GGSN}]
Create PDP Context response
[PDP Address = FE80::ID_{M-GGSN}]
IPV6CP Configure-Nack
[Interface-identifier=ID_{M-GGSN}]
IPV6CP Configure-Request
[Interface-identifier=ID_{M-GGSN}]
IPV6CP Configure-Ack
[Interface-identifier=ID_{M-GGSN}]
Assign FE80::ID_{M-GGSN} to interface
IPv6 router solicitation
IPv6 router advertisement
[set of 64-bit prefixes {P}]
Assign P::ID_{M-GGSN} to interface, for each P

Fig. 5

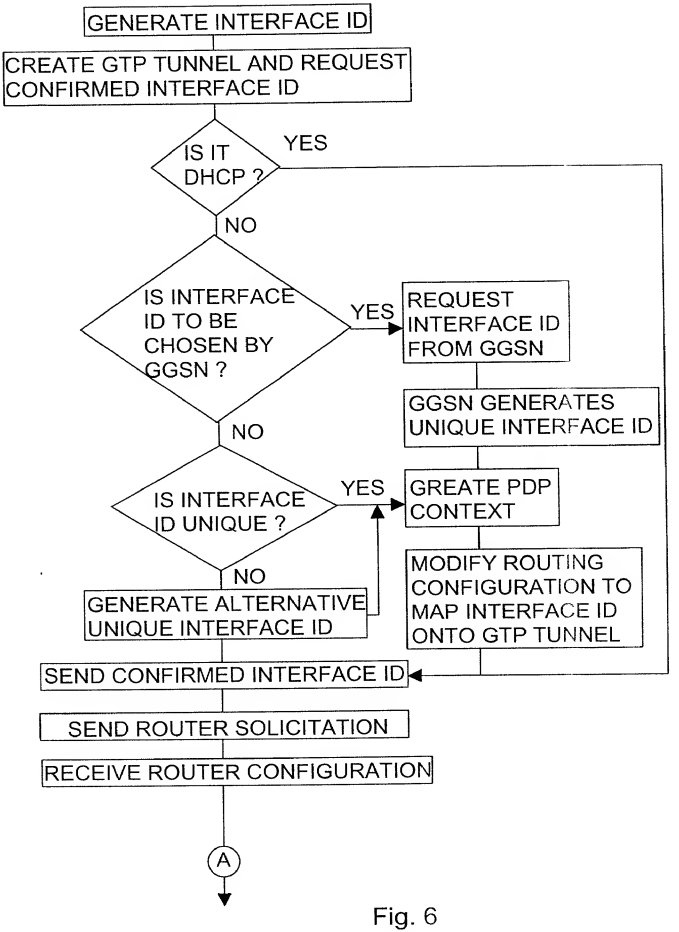
GENERATE INTERFACE ID
CREATE GTP TUNNEL AND REQUEST CONFIRMED INTERFACE ID
IS IT DHCP ?
YES
NO
IS INTERFACE ID TO BE CHOSEN BY GGSN ?
YES
REQUEST INTERFACE ID FROM GGSN
GGSN GENERATES UNIQUE INTERFACE ID
NO
IS INTERFACE ID UNIQUE ?
YES
GREATE PDP CONTEXT
MODIFY ROUTING CONFIGURATION TO MAP INTERFACE ID ONTO GTP TUNNEL
NO
GENERATE ALTERNATIVE UNIQUE INTERFACE ID
SEND CONFIRMED INTERFACE ID
SEND ROUTER SOLICITATION
RECEIVE ROUTER CONFIGURATION
A

Fig. 6

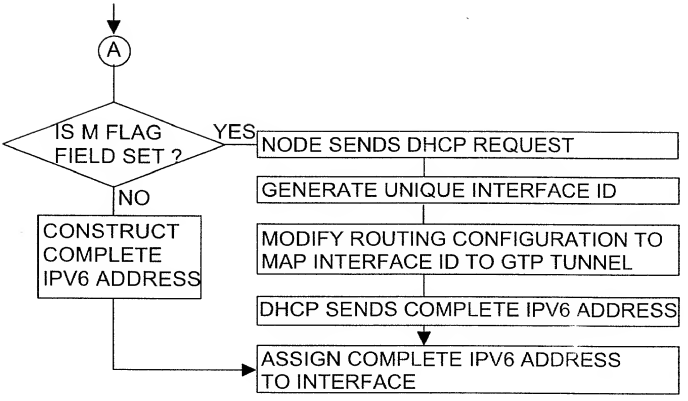
A
IS M FLAG FIELD SET ?
YES
NO
NODE SENDS DHCP REQUEST
GENERATE UNIQUE INTERFACE ID
MODIFY ROUTING CONFIGURATION TO MAP INTERFACE ID TO GTP TUNNEL
DHCP SENDS COMPLETE IPV6 ADDRESS
CONSTRUCT COMPLETE IPV6 ADDRESS
ASSIGN COMPLETE IPV6 ADDRESS TO INTERFACE

Fig. 6

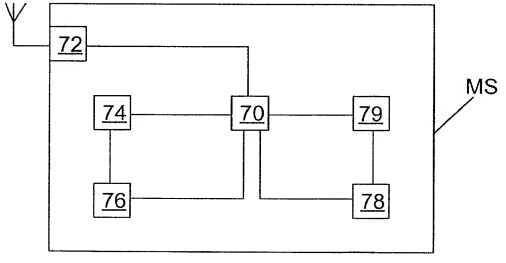
72
74
70
79
76
78
MS

Fig. 8

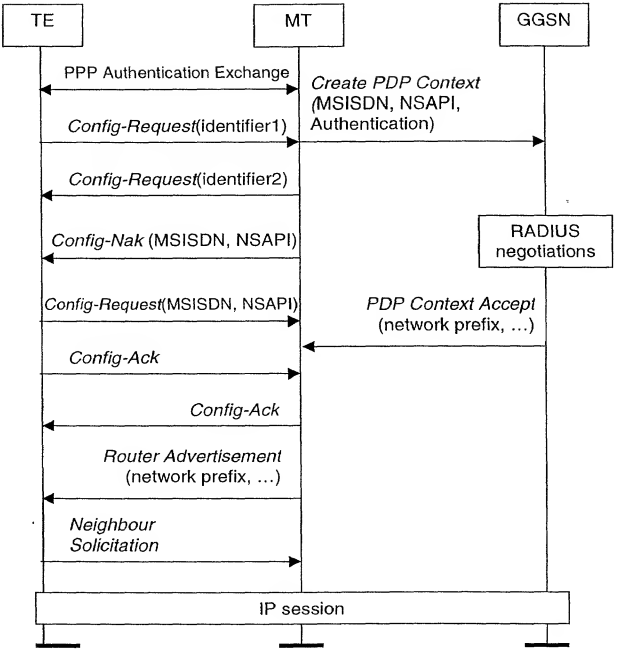
TE
MT
GGSN
PPP Authentication Exchange
Create PDP Context
(MSISDN, NSAPI,
Authentication)
Config-Request(identifier1)
Config-Request(identifier2)
RADIUS
negotiations
Config-Nak (MSISDN, NSAPI)
Config-Request(MSISDN, NSAPI)
PDP Context Accept
(network prefix, ...)
Config-Ack
Config-Ack
Router Advertisement
(network prefix, ...)
Neighbour
Solicitation
IP session

Fig. 7